REMARKS

The foregoing amendment is to impart greater clarity and definiteness to the claims rather than to avoid prior art.

Applicants respectfully request reconsideration of this application as amended. Claims 1-40 are pending in the application. Claims 1-34 are rejected. Claims 35-40 are allowed. Claims 20 and 39 are amended.

Rejections under 35 U.S.C. 101

Claims 1-34 are rejected under 35 U.S.C. 101, as allegedly being directed to non-statutory subject matter. Applicant respectfully disagrees.

Claim 1, for example, sets forth:

> 1. (Original) A method comprising:
> loading a table having a set of L data elements; determining whether said table fits into a single register;
> performing a data lookup into said table with a packed data shuffle operation if said determination indicates that said table does fit into a single register; and
> dividing said table into a plurality of sections if said table does not fit into a single register, each of said sections sized to fit into a single register, and executing a plurality of packed data shuffle operations on said plurality of sections to look up data in said table.

An analysis of the instant claims must be performed in order to make a determination of whether the subject matter is statutory. Such analysis should correlate each claim element with corresponding structures, materials or acts set forth in the specification.

The Federal Circuit makes it clear that the ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention. "The person of ordinary skill in the art is deemed

to read the claim term not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the specification." *Phillips v. AWH Corp.*, 415 F.3d at 1313.

Applicant respectfully submits that the claimed performing a data lookup into a table with a packed data shuffle operation if the table fits into a single register; and dividing the table into sections if the table does not fit into a single register, each section sized to fit into a single register, and executing a plurality of packed data shuffle operations on these sections to look up data in the table, as set forth in Claim 1 (and also in Claim 28), would not be treated merely as a program per se by a person of skill in the art in the context of the entire patent.

Claim 20, as amended, also sets forth:

> 20. (Currently Amended) An article comprising a tangible machine readable medium that stores a program, said program being executable by a machine to perform a method comprising:
> determining whether a table having a set of L data elements fits into a single register;
> performing a data lookup into said table with a packed data shuffle operation if said determination indicates that said table does fit into a single register; and
> dividing said table into a plurality of sections if said table does not fit into a single register, each of said sections sized to fit into a single register, and executing a plurality of packed data shuffle operations on said plurality of sections to look up data in said table.

Applicant respectfully submits that the claimed executing of packed data shuffle operations to look up data in a table according to a program stored on a tangible machine readable medium, as set forth in Claim 20, would not be treated merely as a program per se by a person of skill in the art in the context of the entire patent.

The instant language when correlated with the corresponding structures and processes set forth in the specification (e.g. see Figures 10A-H, 11 and 12; paragraphs 110-122) makes it apparent to one of skill in the art that the claimed invention has practical applications in the technical arts, i.e. to order data from small lookup tables such as may be used in video and encryption applications.

In addition, Applicant respectfully submits, that the present application clearly asserts such a practical application in the technical arts.

For example, paragraph 39 of the specification (emphasis added) asserts that:

> Embodiments of the present invention provide a way to implement a packed byte shuffle instruction with a flush to zero capability as an algorithm that makes use of SIMD related hardware. For one embodiment, the algorithm is based on the concept of shuffling data from a particular register or memory location based on the values of a control mask for each data element position. Embodiments of a packed byte shuffle can be used to reduce the number of instructions required in many different applications that rearrange data. A packed byte shuffle instruction can also be used for any application with unaligned loads. Embodiments of this shuffle instruction can be used for filtering to arrange data for efficient multiply-accumulate operations. Similarly, a packed shuffle instruction can be used in video and encryption applications for ordering data and small lookup tables. This instruction can be used to mix data from two or more registers. Thus embodiments of a packed shuffle with a flush to zero capability algorithm in accordance with the present invention can be implemented in a processor to support SIMD operations efficiently without seriously compromising overall performance.

Paragraph 45 of the specification (emphasis added) asserts that:

> Similarly, the reversing of all the bytes in a 128 bit register, such as in changing between big endian and little endian formats, can be easily performed with a single packed shuffle instruction. Whereas even these fairly simple patterns require a number of instructions if a packed shuffle instruction were not used, complex or random patterns require even more inefficient instruction routines. The most straight forward solution to rearrange random bytes in a SIMD register is to write them to a buffer and then use integer byte reads/writes to rearrange them and read them back into a SIMD register. All these data processing would require a lengthy code sequence, while a single packed shuffle instructions can suffice. By reducing the number of instructions required, the number of clock cycles needed to produce the same result is greatly reduced. Embodiments of the present invention also use shuffle instructions to access multiple values in a table with a SIMD instructions. Even in the case where the a table is twice the size of a register, algorithms in accordance with the present invention allow for accesses to data elements at a faster rate than the one data element per instruction as with integer operations.

Paragraphs 108-109 of the specification (emphasis added) assert that:

> Currently, table lookups using integer instructions requires a large number of instructions. An even greater number of instructions are needed per lookup if integer operations are used to access data for algorithms implemented with SIMD

instructions. But by using embodiments of a packed byte shuffle instruction, the instruction count and execution time is drastically reduced. For instance, sixteen data bytes can be accessed during a table lookup with a single instruction if the table size is sixteen bytes or less. Eleven SIMD instructions can be used to lookup table data if the table size is between seventeen and thirty two bytes. Twenty three SIMD instructions are needed if the table size is between thirty three and sixty four bytes.

There are some applications with data parallelism that cannot be implemented with SIMD instructions due to their use of lookup tables. The quantization and deblocking algorithms of the video compression method H.26L is an example of an algorithm that uses small lookup tables that may not fit into a 128 bit register. In some cases, the lookup tables used by these algorithms are small. If the table can fit in a single register, the table lookup operation can be accomplished with one packed shuffle instruction. But if the memory space requirement of the table exceeds the size of a single register, embodiments of a packed shuffle instruction can still work via a different algorithm. One embodiment of a method for handling oversized tables divides a table into sections, each equal to the capacity of a register, and accesses each of these table sections with a shuffle instruction. The shuffle instruction uses the same shuffle control sequence to access each section of the table. As a result, a parallel table lookup can be implemented in these cases with the packed byte shuffle instruction, thus permitting the use of SIMD instructions to improve algorithm performance. Embodiments of the present invention can help improve performance and reduce the number of memory accesses needed for algorithms that use small lookup tables. Other embodiments also permit access of multiple lookup table elements using SIMD instructions. A packed byte shuffle instruction in accordance to the present invention permits efficient SIMD instruction implementation instead of less efficient integer implantation of algorithms that use small lookup tables. This embodiment of the present invention demonstrates how to access data from a table that requires memory space larger than a single register. In this example, the registers contain different segments of the table

Paragraph 147 of the specification (emphasis added) asserts that:

Embodiments of algorithms using packed shuffle instructions in accordance with the present invention can also improve processor and system performance with present hardware resources. But as technology continues to improve, embodiments of the present invention when combined with greater amounts of hardware resources and faster, more efficient logic circuits, can have an even more profound impact on improving performance. Thus, one efficient embodiment of a packed shuffle instruction having byte granularity and a flush to zero option can have different and greater impact across processor generations. Simply adding more resources in modern processor architectures alone does not guarantee better performance improvement. By also maintaining the efficiency of

applications like one embodiment of the parallel table lookup and the packed byte shuffle instruction (PSHUFB), larger performance improvements can be possible.

Thus the specification makes it readily apparent to one of skill in the art that the claimed invention has a practical application in the technical arts.

The Supreme Court held that the focus in any statutory subject matter analysis be on the claim as a whole, stating "When a claim containing a mathematical formula implements or applies that formula in a structure or process which, when considered as a whole, is performing a function which the patent laws were designed to protect (e.g., transforming or reducing an article to a different state or thing, then the claim satisfies the requirements of § 101." *In re Alappat*, 33 F.3d 1526, 1543 (Fed. Cir. 1994) (quoting *Diehr*, 450 U.S. at 192, 209 USPQ at 10).

This notion is sometimes phrased in terms of requiring a transformation or reduction of 'subject matter.' In Schrader, the phrase 'subject matter' was determined not to be limited to tangible articles or objects, but includes intangible subject matter, such as data or signals, representative of or constituting physical activity or objects. *Schrader*, 22 F.3d at 295; 30 USPQ2D (BNA) at 1459.

Thus Applicant respectfully submits that Claims 1-34 are directed to statutory subject matter.

CONCLUSION

Applicants respectfully submit the amended specification, the amended drawings, and the present claims for allowance. If the Examiner believes a telephone conference would expedite or assist in the allowance of the present application, the Examiner is invited to call Lawrence Mennemeier at (408) 765-2194.

Authorization is hereby given to charge our Deposit Account No. 02-2666 for any charges that may be due.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN

Date: February 20, 2007

Lawrence M. Mennemeier
Reg. No. 51,003

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1026
(408) 720-8300